

Day 4: Classification

Kenneth Benoit and Slava Mikhaylov

Introduction to Data Science and Big Data Analytics

18 August 2016

Week 3 Outline

Classification

Maximum Likelihood

Logistic regression with several variables

Confounding

Logistic regression with more than two classes

Discriminant Analysis

- Bayes theorem for classification

- Linear Discriminant Analysis when $p > 1$

- Characterizing performance of classifiers

- Other forms of Discriminant Analysis

- Naive Bayes Classifier

Logistic Regression versus LDA

Classification

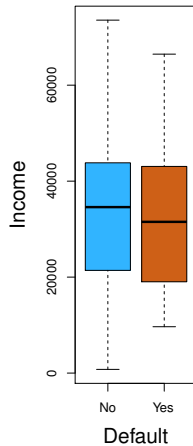
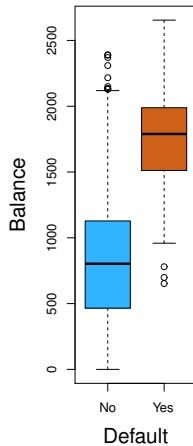
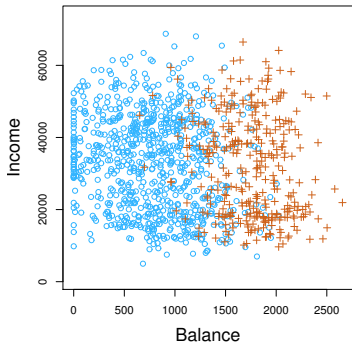
Classification

- ▶ Qualitative variables take values in an unordered set \mathcal{C} , such as: $eye\ color \in \{brown, blue, green\}$; $email \in \{spam, ham\}$.
- ▶ Given a feature vector X and a qualitative response Y taking values in the set \mathcal{C} , the classification task is to build a function $\mathcal{C}(\mathcal{X})$ that takes as input the feature vector X and predicts its value for Y ; i.e. $\mathcal{C}(\mathcal{X}) \in \mathcal{C}$.

Classification

- ▶ Often we are more interested in estimating the **probabilities** that X belongs to each category in \mathcal{C} .
- ▶ For example, it is more valuable to have an estimate of the probability that an insurance claim is fraudulent, than a classification fraudulent or not.

Example: Credit Card Default



Can we use Linear Regression?

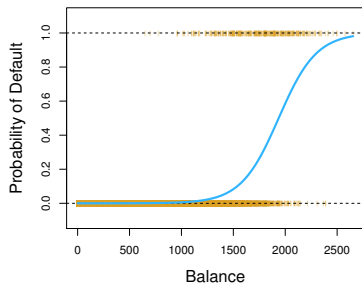
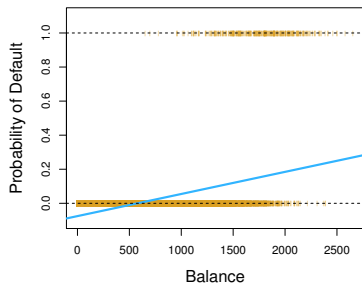
Suppose for the **Default** classification task that we code

$$Y = \begin{cases} 0 & \text{if No} \\ 1 & \text{if Yes.} \end{cases}$$

Can we simply perform a linear regression of Y on X and classify as **Yes** if $\hat{Y} > 0.5$?

- ▶ In this case of a binary outcome, linear regression does a good job as a classifier, and is equivalent to **linear discriminant analysis** which we discuss later.
- ▶ Since in the population $E(Y|X = x) = Pr(Y = 1|X = x)$, we might think that regression is perfect for this task.
- ▶ However, **linear** regression might produce probabilities less than zero or bigger than one. **Logistic regression** is more appropriate.

Linear versus Logistic Regression



- ▶ The orange marks indicate the response Y , either 0 or 1.
- ▶ Linear regression does not estimate $Pr(Y = 1|X)$ well.
- ▶ Logistic regression seems well suited to the task.

Linear Regression continued

- ▶ Now suppose we have a response variable with three possible values. A patient presents at the emergency room, and we must classify them according to their symptoms.

$$Y = \begin{cases} 1 & \text{if } \textit{stroke}; \\ 2 & \text{if } \textit{drug overdose}; \\ 3 & \text{if } \textit{epileptic seizure}. \end{cases}$$

- ▶ This coding suggests an ordering, and in fact implies that the difference between *stroke* and *drug overdose* is the same as between *drug overdose* and *epileptic seizure*.
- ▶ Linear regression is not appropriate here.
- ▶ **Multiclass Logistic Regression** or **Discriminant Analysis** are more appropriate.

Logistic Regression

- ▶ Let's write $p(X) = \Pr(Y = 1|X)$ for short and consider using *balance* to predict *default*. Logistic regression uses the form

$$p(X) = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}}.$$

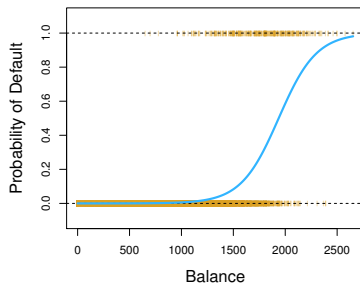
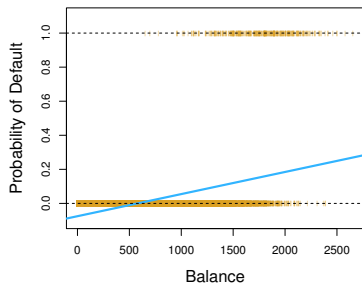
($e \approx 2.71828$ is a mathematical constant [Euler's number.])

- ▶ It is easy to see that no matter what values β_0 , β_1 or X take, $p(X)$ will have values between 0 and 1.
- ▶ A bit of rearrangement gives

$$\log \left(\frac{p(X)}{1 - p(X)} \right) = \beta_0 + \beta_1 X.$$

- ▶ This monotone transformation is called the **log odds** or **logit** transformation of $p(X)$.

Linear versus Logistic Regression



- ▶ Logistic regression ensures that our estimate for $p(X)$ lies between 0 and 1.

Maximum Likelihood

- ▶ We use maximum likelihood to estimate the parameters.

$$\ell(\beta_0, \beta) = \prod_{i: y_i=1} p(x_i) \prod_{i: y_i=0} (1 - p(x_i)).$$

- ▶ This **likelihood** gives the probability of the observed zeros and ones in the data.
- ▶ We pick β_0 and β_1 to maximize the likelihood of the observed data.
- ▶ Most statistical packages can fit linear logistic regression models by maximum likelihood. In R we use the *glm* function.

```
library(ISLR)
data("Default")
names(Default)

## [1] "default" "student" "balance" "income"

logit <- glm(Default$default ~ Default$balance, family = binomial)
```

```
summary(logit)
```

```
##
```

```
## Call:
```

```
## glm(formula = Default$default ~ Default$balance, family = binomial)
```

```
##
```

```
## Deviance Residuals:
```

```
##      Min       1Q   Median       3Q      Max  
## -2.2697  -0.1465  -0.0589  -0.0221   3.7589
```

```
##
```

```
## Coefficients:
```

```
##              Estimate Std. Error z value Pr(>|z|)  
## (Intercept)   -1.065e+01  3.612e-01  -29.49  <2e-16 ***  
## Default$balance  5.499e-03  2.204e-04   24.95  <2e-16 ***
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
```

```
##      Null deviance: 2920.6  on 9999  degrees of freedom
```

```
## Residual deviance: 1596.5  on 9998  degrees of freedom
```

```
## AIC: 1600.5
```

```
##
```

```
## Number of Fisher Scoring iterations: 8
```

Making Predictions

- ▶ What is our estimated probability of *default* for someone with a balance of \$1000?

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}} = \frac{e^{-10.6513 + 0.0055 \times 1000}}{1 + e^{-10.6513 + 0.0055 \times 1000}} = 0.006$$

- ▶ With a balance of \$2000?

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 X}} = \frac{e^{-10.6513 + 0.0055 \times 2000}}{1 + e^{-10.6513 + 0.0055 \times 2000}} = 0.586$$

```

logit2 <- glm(Default$default ~ Default$student, family = binomial)
summary(logit2)

##
## Call:
## glm(formula = Default$default ~ Default$student, family = binomial)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.2970  -0.2970  -0.2434  -0.2434   2.6585
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -3.50413    0.07071  -49.55  < 2e-16 ***
## Default$studentYes  0.40489    0.11502   3.52 0.000431 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 2908.7  on 9998  degrees of freedom
## AIC: 2912.7
##
## Number of Fisher Scoring iterations: 6

```


Making Predictions (binary variable)

$$\widehat{Pr}(\text{default} = \text{Yes} | \text{student} = \text{Yes}) = \frac{e^{-3.5041 + 0.4049 \times 1}}{1 + e^{-3.5041 + 0.4049 \times 1}} = 0.0431$$

$$\widehat{Pr}(\text{default} = \text{Yes} | \text{student} = \text{No}) = \frac{e^{-3.5041 + 0.4049 \times 0}}{1 + e^{-3.5041 + 0.4049 \times 0}} = 0.0292$$

Logistic regression with several variables

$$\log \left(\frac{p(X)}{1 - p(X)} \right) = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p$$

$$p(X) = \frac{e^{\beta_0 + \beta_1 X_1 + \dots + \beta_p X_p}}{1 + e^{\beta_0 + \beta_1 X_1 + \dots + \beta_p X_p}}$$

```

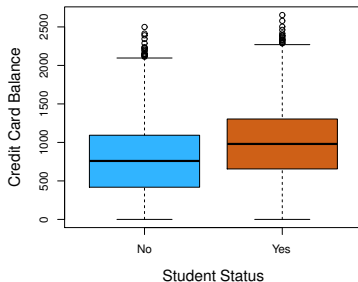
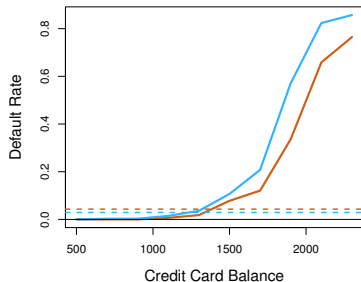
logit3 <- glm(Default$default ~ Default$balance + Default$income + Default$stud
summary(logit3)

##
## Call:
## glm(formula = Default$default ~ Default$balance + Default$income +
##      Default$student, family = binomial)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.4691  -0.1418  -0.0557  -0.0203   3.7383
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.087e+01  4.923e-01 -22.080  < 2e-16 ***
## Default$balance    5.737e-03  2.319e-04  24.738  < 2e-16 ***
## Default$income     3.033e-06  8.203e-06   0.370  0.71152
## Default$studentYes -6.468e-01  2.363e-01  -2.738  0.00619 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1571.5  on 9996  degrees of freedom
## AIC: 1579.5
##
## Number of Fisher Scoring iterations: 8

```

- ▶ Why is coefficient for *student* negative, while it was positive before?

Confounding



- ▶ Students tend to have higher balances than non-students, so their marginal default rate is higher than for non-students.
- ▶ But for each level of balance, students default less than non-students.
- ▶ Multiple logistic regression can tease this out.

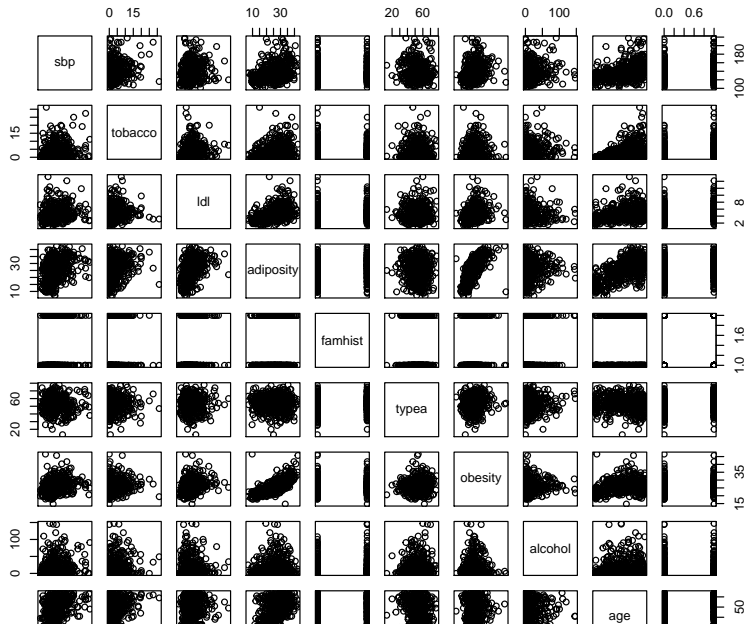
Example: South African Heart Disease

- ▶ 160 cases of MI (myocardial infarction) and 302 controls (all male in age range 15-64), from Western Cape, South Africa in early 80s.
- ▶ Overall prevalence very high in this region: 5.1%.
- ▶ Measurements on seven predictors (risk factors), shown in scatterplot matrix.
- ▶ Goal is to identify relative strengths and directions of risk factors.
- ▶ This was part of an intervention study aimed at educating the public on healthier diets.

```
library(ElemStatLearn)
data("SAheart")
names(SAheart)
```

```
## [1] "sbp"      "tobacco"  "ldl"      "adiposity" "famhist"
## [6] "typea"    "obesity"  "alcohol"  "age"       "chd"
```

`pairs(SAheart)`




```

heartfit <- glm(chd ~ . , data = SAheart, family = binomial)
summary(heartfit)

##
## Call:
## glm(formula = chd ~ ., family = binomial, data = SAheart)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.7781  -0.8213  -0.4387   0.8889   2.5435
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.1507209  1.3082600  -4.701 2.58e-06 ***
## sbp           0.0065040  0.0057304   1.135 0.256374
## tobacco      0.0793764  0.0266028   2.984 0.002847 **
## ldl           0.1739239  0.0596617   2.915 0.003555 **
## adiposity     0.0185866  0.0292894   0.635 0.525700
## famhistPresent 0.9253704  0.2278940   4.061 4.90e-05 ***
## typea         0.0395950  0.0123202   3.214 0.001310 **
## obesity      -0.0629099  0.0442477  -1.422 0.155095
## alcohol       0.0001217  0.0044832   0.027 0.978350
## age          0.0452253  0.0121298   3.728 0.000193 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##

```

Logistic regression with more than two classes

- ▶ So far we have discussed logistic regression with two classes.
- ▶ It is easily generalized to more than two classes.
- ▶ One version (used in the R package *glmnet*) has the symmetric form

$$Pr(Y = k|X) = \frac{e^{\beta_{0k} + \beta_{1k}X_1 + \dots + \beta_{pk}X_p}}{\sum_{\ell=1}^K e^{\beta_{0\ell} + \beta_{1\ell}X_1 + \dots + \beta_{p\ell}X_p}}$$

- ▶ Here there is a linear function for **each** class.
- ▶ Multiclass logistic regression is also referred to as **multinomial regression**.

Discriminant Analysis

- ▶ Here the approach is to model the distribution of X in each of the classes separately, and then use **Bayes theorem** to flip things around and obtain $Pr(Y|X)$.
- ▶ When we use normal (Gaussian) distributions for each class, this leads to linear or quadratic discriminant analysis.
- ▶ However, this approach is quite general, and other distributions can be used as well. Here, we will focus on normal distributions.

Bayes theorem for classification

- ▶ Thomas Bayes was a famous mathematician whose name represents a big subfield of statistical and probabilistic modeling.
- ▶ Here we focus on a simple result, known as Bayes theorem:

$$Pr(Y = k|X = x) = \frac{Pr(X = x|Y = k) \cdot Pr(Y = k)}{Pr(X = x)}$$

- ▶ One writes this slightly differently for discriminant analysis:

$$Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^K \pi_l f_l(x)},$$

where

- ▶ $f_k(x) = Pr(X = x|Y = k)$ is the **density** for X in class k . Here we will use normal densities for these, separately in each class.
- ▶ $\pi_k = Pr(Y = k)$ is the marginal or **prior** probability for class k .

Why discriminant analysis?

- ▶ When the classes are well-separated, the parameter estimates for the logistic regression model are surprisingly unstable. Linear discriminant analysis does not suffer from this problem.
- ▶ If n is small and the distribution of the predictors X is approximately normal in each of the classes, the linear discriminant model is again more stable than the logistic regression model.
- ▶ Linear discriminant analysis is popular when we have more than two response classes, because it also provides low-dimensional views of the data.

Linear Discriminant Analysis when $p = 1$

- ▶ The Gaussian density has the form

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma_k}\right)^2}$$

- ▶ Here μ_k is the mean, and σ_k^2 the variance (in class k). We will assume that all the $\sigma_k = \sigma$ are the same.
- ▶ Plugging this into Bayes formula, we get a rather complex expression for $p_k(x) = Pr(Y = k|X = x)$:

$$f_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}{\sum_{\ell=1}^K \pi_\ell \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_\ell}{\sigma}\right)^2}}$$

- ▶ There are simplifications and cancellations.

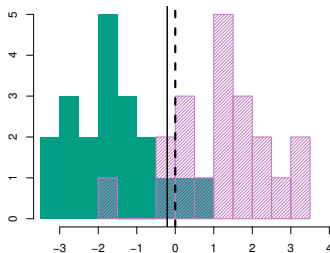
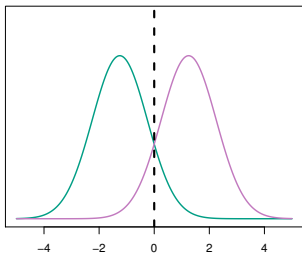
Discriminant functions

- ▶ To classify at the value $X = x$, we need to see which of the $p_k(x)$ is largest. Taking logs, and discarding terms that do not depend on k , we see that this is equivalent to assigning x to the class with the largest discriminant score:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

- ▶ Note that $\delta_k(x)$ is a **linear** function of x .
- ▶ If there are $K = 2$ classes and $\pi_1 = \pi_2 = 0.5$, then one can see that the **decision boundary** is at

$$x = \frac{\mu_1 + \mu_2}{2}.$$

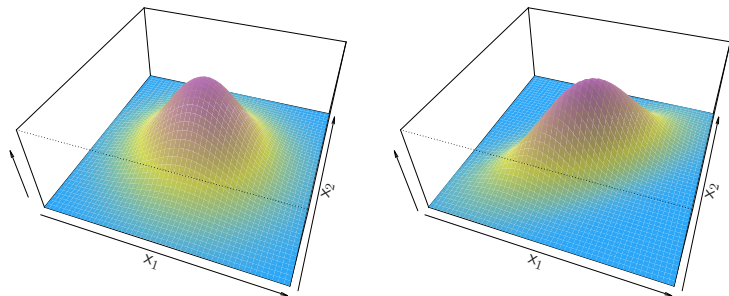


- ▶ Example with $\mu_1 = -1.5$, $\mu_2 = 1.5$, $\pi_1 = \pi_2 = 0.5$, and $\sigma^2 = 1$.
- ▶ Typically we don't know these parameters; we just have the training data.
- ▶ In that case we simply estimate the parameters and plug them into the rule.

Estimating the parameters

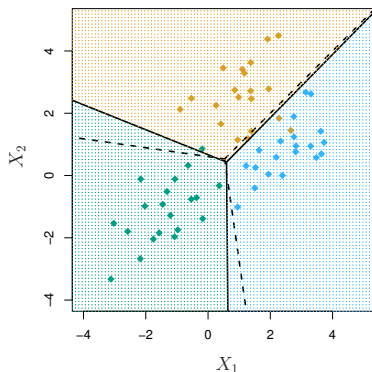
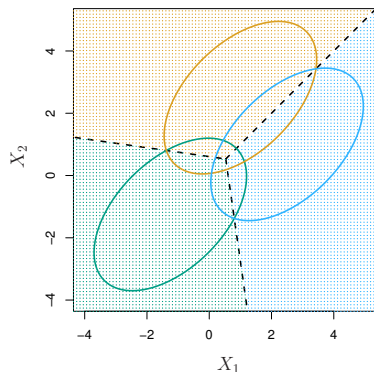
- ▶ $\hat{\pi}_k = \frac{n_k}{n};$
- ▶ $\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i;$
- ▶ $\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^K \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2 = \sum_{k=1}^K \frac{n_k-1}{n-K} \cdot \hat{\sigma}_k^2$
- ▶ where $\hat{\sigma}_k^2 = \frac{1}{n_k-1} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$ is the usual formula for the estimated variance in the k th class.

Linear Discriminant Analysis when $p > 1$



- Density: $f(x) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$
- Discriminant function:
$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log \pi_k$$
- Despite its complex form,
$$\delta_k(x) = c_{k0} + c_{k1}x_1 + c_{k2}x_2 + \cdots + c_{kp}x_p$$
 – a linear function.

Illustration: $p = 2$ and $K = 3$ classes



- ▶ Here $\pi_1 = \pi_2 = \pi_3 = 1/3$.
- ▶ The dashed lines are known as the **Bayes decision boundaries**.
- ▶ Were they known, they would yield the fewest misclassification errors, among all possible classifiers.

From $\delta_k(x)$ to probabilities

- Once we have estimates $\delta_k(x)$, we can turn these into estimates for class probabilities:

$$\widehat{Pr}(Y = k|X = x) = \frac{e^{\hat{\delta}_k(x)}}{\sum_{\ell=1}^K e^{\hat{\delta}_\ell(x)}}.$$

- So classifying to the largest $\hat{\delta}_k(x)$ amounts to classifying to the class for which $\widehat{Pr}(Y = k|X = x)$ is largest.
- When $K = 2$, we classify to class 2 if $\widehat{Pr}(Y = 2|X = x) \geq 0.5$, else to class 1.

Confusion matrix and error rates

| | | True No | Default Yes | Status Total |
|----------------|-----|------------|----------------|-----------------|
| Predicted | No | 9644 | 252 | 9896 |
| Default Status | Yes | 23 | 81 | 104 |
| Total | | 9667 | 333 | 10000 |

- ▶ $(23 + 252) / 10000$ errors — a 2.75% misclassification rate.
- ▶ Some caveats:
 - ▶ This is **training** error, and we may be overfitting. Not a big concern here since $n = 10000$ and $p = 4$.
 - ▶ If we classified to the prior – always to class *No* in this case – we would make $333/10000$ errors, or only 3.33%.
 - ▶ Of the true *No*'s, we make $23/9667 = 0.2\%$ errors; of the true *Yes*'s, we make $252/333 = 75.7\%$ errors!

Types of errors

- ▶ **False positive rate:** The fraction of negative examples that are classified as positive – 0.2% in example.
- ▶ **False negative rate:** The fraction of positive examples that are classified as negative – 75.7% in example.

Sensitivity and specificity

- ▶ Performance of a classifier is often characterized in terms of **sensitivity** and **specificity**.
- ▶ Here, the sensitivity is the percentage of true defaulters that are identified. It is 24.3% in our case.
- ▶ The specificity is the percentage of non-defaulters that are correctly identified. Here it is $(1 - 23/9,667) \cdot 100 = 99.8\%$
- ▶ The true positive rate is the sensitivity of our classifier.
- ▶ The false positive rate is *one minus* the specificity of our classifier.

Errors and threshold

- ▶ We produced the confusion matrix above by classifying to class Yes if

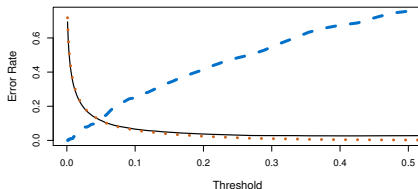
$$\widehat{Pr}(\text{Default} = \text{Yes} | \text{Balance}, \text{Student}) \geq 0.5$$

- ▶ We can change the two error rates by changing the threshold from 0.5 to some other value in $[0,1]$:

$$\widehat{Pr}(\text{Default} = \text{Yes} | \text{Balance}, \text{Student}) \geq \text{threshold},$$

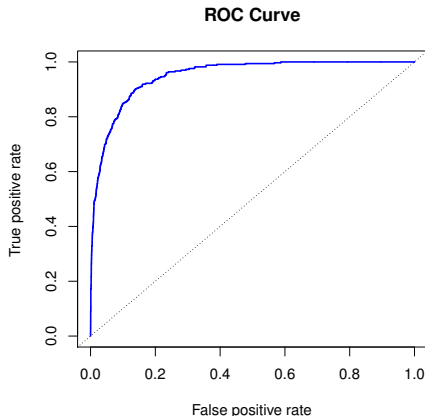
and vary *threshold*.

Varying the *threshold*



- ▶ Error rates are shown as a function of the threshold value for the posterior probability that is used to perform the assignment.
- ▶ The black solid line displays the overall error rate.
- ▶ The blue dashed line represents the fraction of defaulting customers that are incorrectly classified (**False Negative**).
- ▶ The orange dotted line indicates the fraction of errors among the non-defaulting customers (**False Positive**).
- ▶ In order to reduce the false negative rate, we may want to reduce the threshold to 0.1 or less.

ROC curve



- ▶ The ROC plot displays both simultaneously.
- ▶ Sometimes we use the AUC or area under the curve to summarize the overall performance.
- ▶ Higher AUC is good.

Characterizing performance of classifiers

| | | Predicted class | | |
|------------|---------------|-----------------|----------------|-------|
| | | - or Null | + or Non-null | Total |
| True class | - or Null | True Neg. (TN) | False Pos.(FP) | N |
| | + or Non-null | False Neg. (FN) | True Pos. (TP) | P |
| Total | | N* | P* | |

- ▶ "+" is "disease" or alternative (non-null) hypothesis (here, those who default);
- ▶ "-" is "non-disease" or the null hypothesis (here, those who do not default).

Performance measures for classifiers

| Name | Definition | Synonyms |
|------------------|------------|---|
| False Pos. rate | FP/N | Type I error, 1- Specificity |
| True Pos. rate | TP/P | 1 - Type II error, power, sensitivity, recall |
| Pos. Pred. value | TP/P^* | Precision, 1-false discovery proportion |
| Neg. Pred. value | TN/N^* | |

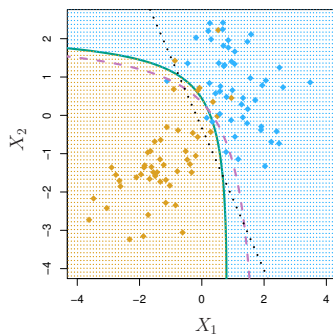
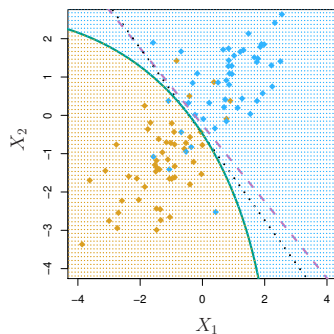
- ▶ The denominators for the false positive and true positive rates are the actual population counts in each class.
- ▶ The denominators for the positive predictive value and the negative predictive value are the total predicted counts for each class.

Other forms of Discriminant Analysis

$$Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{\ell=1}^K \pi_{\ell} f_{\ell}(x)}$$

- ▶ When $f_k(x)$ are Gaussian densities, with the same covariance matrix Σ in each class, this leads to linear discriminant analysis.
- ▶ By altering the forms for $f_k(x)$, we get different classifiers.
 - ▶ With Gaussians but different Σ_k in each class, we get **quadratic discriminant analysis**.
 - ▶ With $f_k(x) = \prod_{j=1}^p f_{jk}(x_j)$ (conditional independence model) in each class we get **naive Bayes**. For Gaussian this means the Σ_k are diagonal.
 - ▶ Many other forms, by proposing specific density models for $f_k(x)$, including nonparametric approaches.

Quadratic Discriminant Analysis



- ▶ $\delta_k(x) = -\frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1}(x - \mu_k) + \log \pi_k$
- ▶ Because the Σ_k are different, the quadratic terms matter.

Naive Bayes

- ▶ Assumes features are independent in each class.
- ▶ Useful when p is large, and so multivariate methods like QDA and even LDA break down.
- ▶ Gaussian naive Bayes assumes each Σ_k is diagonal:

$$\delta_k(x) \propto \log \left[\pi_k \prod_{j=1}^p f_{kj}(x_j) \right] = -\frac{1}{2} \sum_{j=1}^p \frac{(x_j - \mu_{kj})^2}{\sigma_{kj}^2} + \log \pi_k$$

- ▶ Can use for **mixed** feature vectors (qualitative and quantitative). If X_j is qualitative, replace $f_{kj}(x_j)$ with probability mass function (histogram) over discrete categories.
- ▶ Despite strong assumptions, naive Bayes often produces good classification results.

Naive Bayes Classifier

- ▶ Naive Bayes (NB) classifier especially appropriate when the dimension p of the feature space is high, making density estimation unattractive.
- ▶ Assumes that given a class $G = j$, the features X_k are independent:

$$f_j(X) = \prod_{k=1}^p f_{jk}(X_k).$$

- ▶ While this assumption is pretty heroic and generally not true, it significantly simplifies the estimation.
- ▶ The individual class-conditional marginal densities f_{jk} can each be estimated separately.
- ▶ If a component X_j of X is discrete, then an appropriate histogram estimate can be used. This provides a seamless way of mixing variable types in a feature vector.

Naive Bayes Classifier

- ▶ Despite these strong assumptions, NB classifiers often outperform far more sophisticated alternatives.
- ▶ Although the individual class density estimates may be biased, this bias might not hurt the posterior probabilities as much, especially near the decision regions.
- ▶ In fact, the problem may be able to withstand considerable bias for the savings in variance such a “naive” assumption earns.

Logistic Regression versus LDA

- ▶ For a two-class problem, one can show that for LDA

$$\log \left(\frac{p_1(x)}{1 - p_1(x)} \right) = \log \left(\frac{p_1(x)}{p_2(x)} \right) = c_0 + c_1 x_1 + \cdots + c_p x_p$$

- ▶ So it has the same form as logistic regression.
- ▶ The difference is in how the parameters are estimated.
 - ▶ Logistic regression uses the conditional likelihood based on $Pr(Y|X)$ (known as **discriminative learning**).
 - ▶ LDA uses the full likelihood based on $Pr(X, Y)$ (known as **generative learning**).
 - ▶ Despite these differences, in practice the results are often very similar.
- ▶ Note: logistic regression can also fit quadratic boundaries like QDA, by explicitly including quadratic terms in the model.

Summary

- ▶ Logistic regression is very popular for classification, especially when $K = 2$.
- ▶ LDA is useful when n is small, or the classes are well separated, and Gaussian assumptions are reasonable. Also when $K > 2$.
- ▶ Naive Bayes is useful when p is very large.
- ▶ See Section 4.5 for some comparisons of logistic regression, LDA and KNN.